## REMARKS

Claims 1-5 and 20-35 will be pending in the current Application upon entering this Amendment. Claims 1-5, 20, and 25 have been amended; claims 6-11 and 16-19 have been cancelled (note that claims 12-15 have been previously cancelled), and claims 29-35 have been added. All the amendments herein have been made in order to clarify the claims and not for prior art reasons. Applicants also submit that (1) no amendment made was related to the statutory requirements of patentability unless expressly stated herein, and (2) no amendment made was for the purpose of narrowing the scope of any claim, unless Applicants have argued herein that such amendment was made to distinguish over a particular reference or combination of references.

## Rejection of claims 1-5 under 35 U.S.C. 103

Applicants respectfully submit that claims 1 and 3-5 are patentable under 35 U.S.C. 103 over Stoodley, "Software Pipelining Loops with Conditional Branches" in view of Albert et al., "Data Parallel Computers and the FORALL Statement."

With respect to claim 1, Applicants submit that claim 1 is patentable over Stoodley since Stoodley does not teach or suggest each and every element of claim 1. For example, claim 1 includes "an instruction fetching mechanism that retrieves the set of instructions" where at least one of the set of instructions comprises "a single instruction that provides information corresponding to at least two different loops, wherein the at least two different loops have at least one of different start addresses or different end addresses" The Examiner states that Stoodley addresses a single looping construct but lacks teaching "multiple looping constructs." However, the Examiner proceeds to state that "Stoodley teaches about single VLIW loop instruction that is likely executing from different iterations..., where the instruction VLIW combines many operations, some of operation perform iteration." However, the fact that the VLIW instruction may be performing operations of different iterations is irrelevant to the elements of claim 1. That is, the elements of claim 1 are directed to a single processor instruction (i.e. an instruction fetched by an instruction fetching mechanism and executed by an execution unit) with provides information corresponding to at least two different loops. These two different loops cannot

simply be different iterations of a same loop, as disclosed in Stoodley. That is, the at least two different loops are *different* loops where at least the starting address or ending address of the different loops differ. Therefore, the two different loops are not simply different iterations of a loop. For example, as seen on pg. 263 of Stoodley, even though each loop iteration may take different paths of the if-then statement, the starting loop address (corresponding to the fslt.s instruction) and the ending loops address (corresponding to the bnez.d instruction) are the same. Therefore, the existence of multiple iteration paths does not teach or suggest a single processor instruction which provides information corresponding to at least two different loops. That is, it is irrelevant how many iteration paths may exist within a same loop construct.

The Examiner then combines Stoodley with Albert by stating that "it would have been obvious to a person of ordinary skill in the art at the time of invention was made to modify the operations of a single loop "VLIW instruction" provided with multiple iteration paths" in accordance with the "FORALL" instruction of Albert. However, as discussed above, the VLIW instruction provided with multiple iteration paths is irrelevant to a *single instruction* which provides information for *at least two different loops*. Multiple iteration paths do not teach or suggest different loops. Also, while the FORALL instruction, when compiled, would generate multiple loop constructs, the FORALL instruction would *not* result in a single processor instruction which provides information for at least two different loops. Furthermore, there is no motivation in Stoodley to even address the concept of multiple loops because Stoodley only addresses VLIW instructions used within *a single loop*. Therefore, none of the cited references, alone or in combination, teach or suggest the single instruction as claimed in claim 1.

With respect to claim 2, the Examiner cites Tran (US6,003,128) as teaching the additional limitation because Tran discloses that "a loop count is typical set by initializing a specified register prior to the execution of the loop." However, claim 2 claims that the single instruction provides information to initialize at least two different loops for later execution. Tran adds nothing to Stoodley and Albert to teach or suggest a single instruction providing information corresponding to at least two different loops or for initializing at least two different loops for later execution. Therefore, for at least those reasons discussed above with respect to claim 1, and for these additional reasons, Applicants submit that claim 2 is allowable over the cited references. Applicants submit that claims 3-5 which depend directly or indirectly from allowable claim 1 are also allowable for at least those reasons provided with respect to claim 1.

With respect to claim 4, Applicants have clarified claim 4 such that the single instruction provides at least one loop termination condition corresponding to at least one of the at least two different loops. The Examiner states that with respect to Stoodley in which the VLIW instructions combine operations, "some of these operation perform multiple iterations and dealt with condition branches." However, as stated above, a single VLIW instruction which may perform operations from multiple loop iterations is irrelevant to the single instruction claimed in claim 1. Furthermore, the conditional branches within the loop of Stoodley does not teach or suggest a termination condition for the loop. Therefore, for these additional reasons, Applicants submit that claim 4 is allowable over the cited references.

**Rejection of claims 20-28 under 35 U.S.C. 103**

Applicants respectfully submit that claims 20-28 are patentable under 35 U.S.C. 103 over Stoodley in view of Albert and further in view of Tran.

With respect to claim 20, none of the cited references teach a single instruction as claimed in claim 20. As discussed above, none of the cited references teach or suggest "in response to decoding the single instruction, using information provided by the single instruction to initialize a plurality of loop storage elements corresponding to the first loop and the second loop." That is, the VLIW instructions in Stoodley are described in reference to multiple iteration paths of a single loop but Stoodley makes no mention of a single instruction providing information to initialize loop storage elements corresponding to both a first loop and a second loop. The first loop and the second loop are *not* simply iterations of *a same loop,* as described in Stoodley, but represent multiple looping constructs. Furthermore, note that the single instruction of claim 20 (and of claim 1 above) is a processor instruction (i.e. it is fetched and decoded by a processor) and thus is not simply a high level operation. That is, the Fortran "forall" operation in Albert are not processor instructions. Therefore, in Albert, multiple processor instructions, rather than a single instruction, would have to be fetched and decoded in order to perform those operations. Therefore, the combination of Stoodley and Albert also does not result in the single instruction of claim 20. Also, Tran does not add anything to Stoodley and Albert which result in a *single* instruction which provides information for *multiple* loop constructs.

The Examiner concludes that it would have been obvious to include a multiple loop of Albert at high level programming into the execution mechanism as disclosed by Stoodley and the teaching initializing of Tran. The Examiner states that "doing so, when converting a high level multiple loop instruction (for example, FORALL) into a low level single instruction of VLIWs, would result a single instruction VLIW provided for execution of other instruction of the set of instruction in according with multiple looping constructs...." However, as described above, there is no teaching or suggestion in Albert or Stoodley of converting a high level instruction into a single low level instruction which provides *for multiple looping constructs*. Furthermore, none of the VLIW instructions in Stoodley, as described in detail above with respect to claim 1, teach or suggest initializing multiple looping constructs (i.e. a first loop and a second loop). Therefore, for at least these reasons, Applicants submit that claim 20 is allowable over the cited references. Claims 21-28 depend directly or indirectly from allowable claim 20 and are therefore also allowable for at least those reasons provided with respect to claim 20.

With respect to claims 22 and 25, the Examiner states that claim 25 has the "similar functionality as recited in claim 22" and therefore rejects claim 25 for the same reasons set forth with respect to claim 22. Firstly, with respect to claim 22, Tran does not teach or suggest storing at least one termination condition for each of a first loop *and* a second loop. Furthermore, as discussed above, none of the cited reference teach or suggest a single instruction which initializes storage elements for at least two different loops, such as the first loop and second loop claimed in claim 20. Secondly, the elements of claim 25 do not have similar functionality. Applicants have amended claim 25 to simply clarify that the indicator is an address. That is, the functionality of the ending address of a loop is not the same as termination conditions for a loop. Therefore, their functionality is not similar, and Tran does not teach or suggest storing and ending address of a loop or of different loops, as claimed in claim 25. Therefore, for these additional reasons, Applicants submit that claims 22 and 25 are allowable over the cited references.

**Added Claims 29-35**

Applicants respectfully submit that added claims 29-35 are patentable over the cited references. Claims 29-32 depend, directly or indirectly, from allowable claim 1, and are

therefore allowable for at least those reasons provided above with respect to claim 1. Claims 33-35 depend, directly or indirectly, from allowable claim 20, and are therefore allowable for at least those reasons provided above with respect to claim 20. Furthermore, each of added claims 29-35 provide elements which are also not taught or suggested by any of the cited references, alone or in combination.

## Conclusion

Although Applicants may disagree with statements made by the Examiner in reference to the claims and the cited references, Applicants are not discussing all these statements in the current Office Action, yet reserve the right to address them at a later time if necessary.

Applicants respectfully solicit allowance of the pending claims. Contact me if there are any issues regarding this communication or the current Application.

If Applicant has overlooked any additional fees, or if any overpayment has been made, the Commissioner is hereby authorized to credit or debit Deposit Account 503079, Freescale Semiconductor, Inc.

Respectfully submitted,

By: _____

Joanna G. Chiu
Attorney of Record
Reg. No.: 43,629
Telephone: (512) 996-6839
Fax No.: (512) 996-6854
Email: Joanna.Chiu@Motorola.com

SEND CORRESPONDENCE TO:
Freescale Semiconductor, Inc.
A Motorola Subsidiary
Law Department

Customer Number: 23125